

A Novel Approach for Neighbourhood-based Collaborative Filtering

*Thesis submitted in partial fulfilment
of the requirements for the degree of*

Bachelor of Technology

in

Computer Science and Engineering

by

Nitesh Agrawal
(Roll: 111CS0117)

under the supervision of

Dr. Korra Sathya Babu

and

Dr. Bidyut Kumar Patra

NIT Rourkela



Department of Computer Science and Engineering
National Institute of Technology Rourkela
Rourkela-769 008, Orissa, India
May, 2015

Declaration

This thesis is a presentation of my original research work. Wherever contributions of others are involved, every effort is made to indicate this clearly, with due reference to the literature, and acknowledgement of collaborative research and discussions. I hereby declare that this thesis is my own work and effort and that it has not been submitted anywhere for any award. The interpretations put forth are based on my reading and understanding of the original texts and they are not published anywhere in the form of books, monographs or articles. The other books, articles and websites, which I have made use of are acknowledged at the respective place in the text. For the present thesis, which I am submitting to NIT Rourkela, no degree or diploma or distinction has been conferred on me before, either in this or in any other University. I bear all responsibility and prosecution for any of the unfair means adopted by me in submitting this thesis.

Nitesh Agrawal
(111CS0177)
NIT, Rourkela.

Acknowledgement

I take this opportunity to express my gratitude to my guide Dr. Korra Sathya Babu for his guidance and motivation throughout the course of this project. I convey my regards to Dr. Bidyut Kumar Patra without whose technical guidance this project would have been a distant aim. I would like to thank my parents for supporting me and showing confidence in me whenever I lost hope. I express my profound gratitude to Almighty and my sisters for their support without which this task could have never been accomplished.

Nitesh Agrawal

Contents

Declaration	i
Acknowledgement	ii
1 Introduction	3
1.1 Types of Collaborative Filtering	3
1.1.1 Matrix Factorization	3
1.1.2 Neighbourhood approach	4
1.1.3 Challenges in collaborative filtering	5
1.1.4 Problem Statement	5
1.1.5 Contribution	5
1.1.6 Organization of Thesis	6
2 Literature Survey	7
2.1 Methods of Similarity calculation	8
2.1.1 Pearson Correlation	8
2.1.2 Cosine-based Similarity	8
2.1.3 Adjusted Cosine	9
2.1.4 Similarity based on singularities	9
2.1.5 Similarity between users taking item based similarity as weight	9
2.2 Methods of prediction calculation	10
3 Proposed Work	12
4 Results and Analysis	14
4.1 Evaluation Metrics	14
4.2 Data-sets Used	15
4.3 Observations	15
4.3.1 Method 1 (without averaging)	15
4.3.2 Method 2 (with averaging)	22
4.4 Discussion	29
5 Conclusion and Future Scope	30

List of Figures

4.1	Mean absolute error vs number of neighbours for 100K data set.	16
4.2	Mean absolute error vs number of neighbours for 1M data set.	16
4.3	MAE for good items vs number of neighbours for 100K dataset	17
4.4	MAE for good items vs number of neighbours for 1M dataset	17
4.5	RMSE vs number of neighbours for 100K dataset	18
4.6	RMSE vs number of neighbours for 1M dataset	18
4.7	Precision vs number of neighbours for 100K dataset	19
4.8	Precision vs number of neighbours for 1M dataset	19
4.9	RECALL vs number of neighbours for 100K dataset	20
4.10	RECALL vs number of neighbours for 1M dataset	20
4.11	F1-score vs number of neighbours for 100K dataset	21
4.12	F1-score vs number of neighbours for 1M dataset	21
4.13	Mean absolute error vs number of neighbours for 100K data set.	22
4.14	Mean absolute error vs number of neighbours for 1M data set.	23
4.15	MAE for good items vs number of neighbours for 100K dataset	24
4.16	MAE for good items vs number of neighbours for 1M dataset	24
4.17	RMSE vs number of neighbours for 100K dataset	25
4.18	RMSE vs number of neighbours for 1M dataset	25
4.19	Precision vs number of neighbours for 100K dataset	26
4.20	Precision vs number of neighbours for 1M dataset	26
4.21	RECALL vs number of neighbours for 100K dataset	27
4.22	RECALL vs number of neighbours for 1M dataset	27
4.23	F1-score vs number of neighbours for 100K dataset	28
4.24	F1-score vs number of neighbours for 1M dataset	28

Abstract

Recommender systems hold an integral part in online marketing. It plays an important role for the websites that provide the users an environment to rate and review the products. Several methods can be used to make recommender systems, like content-based filtering, collaborative filtering [1], hybrid approach, which combines content-based as well as collaborative filtering. Collaborative filtering is the most widely used technique to deal with recommender systems. Matrix factorization and neighbourhood approach are the techniques that can be used while dealing with collaborative filtering. Both the methods depends on the ratings that the user has provided in the past. Here we concentrate on neighbourhood approach.

Neighbourhood approach depends on the similarity between items [4] or similarity between users [5], depending on which prediction for an unrated item can be made. The similarity between users or similarity between items can be computed to provide recommendations. Some of the widely used techniques are the Pearson correlation, cosine-based similarity, adjusted cosine, etc. In this thesis a new approach to find similarity between items is used, here the similarity between items is calculated using a modified singularity measure. In this approach, the singularity of ratings provided by each user is taken into consideration [2]. By, using this method recommendation can be found with greater efficiency compared to other existing algorithms as this technique uses the contextual information present in the data.

Keywords: Collaborative filtering; similarity; singularity; prediction .

Notations

$R_{u,i}$ = rating provided by user u to item i .

\bar{R}_u = Average of the ratings provided by user u .

\bar{R}_i = Average of the ratings provided to item i .

I = Column vector of ratings of item i .

$similarity(i, j)$ = similarity between items i and j .

$similarity(u1, u2)$ = similarity between users $u1$ and $u2$.

$P_{u,i}$ = predicted rating for user u on item i .

S_p^x = singulairty of positive ratings provided by user x .

S_n^x = singulairty of negative ratings provided by user x .

Chapter 1

Introduction

Recommendation systems changed the way in which websites interact with users. Rather than having a static experience, in which user searches for a product manually and then buy it, recommender system automates the process. Recommender system automatically provides the user options to buy products, based on their past choice. As the number of users and items are increasing exponentially, an appropriate algorithm is required which could provide recommendations with greater accuracy.

s Collaborative filtering is one among the techniques that are used for this purpose. It can be done using matrix factorization or using neighbourhood approach. Each of these processes requires past ratings provided by the users.

1.1 Types of Collaborative Filtering

There are generally two types of methods that are used for collaborative filtering i.e matrix factorization and neighbourhood approach.

1.1.1 Matrix Factorization

It is a method based on the principle of extraction of latent features underlying the interaction of user and items [11]. For example, two users give high rating to a particular book if the book is a fiction novel, or they like the writer of the book. Hence, if these latent features are discovered, predicting rating for the user for any particular item can be easily carried out, because the features associated with the users must match with the features associated with the item. While discovering different features, it is assumed that number of

features are less than the number of users and number of items. It might suffer from the cold start problem, if the user-item rating matrix is sparse then discovering the features might be difficult.

1.1.2 Neighbourhood approach

This approach is based on finding the neighbours of an item (item-based), or a user (user-based) to get the prediction for an unrated items. To find the neighbours, similarity between items, or users needs to be calculated. It is based on the principle that similar users rate the items similarly, or similar items are rated similarly.

User Based Collaborative Filtering

It is based on the concept of like-minded users [3, 5]. In this approach, ratings provided by users are studied and the pattern is then compared to find similarity between them. It is assumed that like-minded users would rate the item similarly. Therefore, finding the neighbours of each user is the aim of this approach. After the neighbours have been found out, their similarity is used in prediction of the unrated items for the particular user. This method suffers from drawback as, number of users increases exponentially therefore identifying neighbours for each user might require a great deal of computation, which would make the process of similarity calculation too slow and even less efficient.

Item Based Collaborative Filtering

It is based on considering the similarity between items [1, 4]. Items are said to be similar if they have been voted similarly by the different set of users. It looks for the collection of items that the user might have rated in the past, and then the comparison is made to find out how the unrated items are similar to these rated items using various similarity measures. After the similarity between items has been found out, the prediction is made. An advantage of this approach is number of growing users will not affect the efficiency of this approach to a greater extent, and even then less computation will need as compared to user-based approach.

1.1.3 Challenges in collaborative filtering

- Collaborative filtering algorithms work using the past ratings provided by the users, but the user-item matrix provided is large and is sparse one. The sparsity of the matrix may lead to the cold start problem, where the sufficient amount of past rating by the user is unavailable to predict the rating for any other items, thereby leading to irrelevant predictions [1].
- As the number of users and items increases, the collaborative filtering algorithms suffer from scalability issues. The complexity of algorithms is high, and it becomes difficult to handle the huge set of data, thereby increasing the demand for cluster computing. Separate mapper and reducer programs can be made to scale the algorithm, maintaining its efficiency.
- There is a tendency among people to give high rating to their own items and provide low rating to others; it causes a major blunder in recommender systems that use collaborative filtering.

1.1.4 Problem Statement

The goal is to predict the ratings for the items that user has not rated and to achieve this goal, similarity between items are calculated and prediction is done, so that the recommendation can be provided with greater efficiency as compared to traditional methods like, Pearson correlation, cosine-based and, adjusted cosine.

1.1.5 Contribution

This thesis focuses on neighbourhood approach of collaborative filtering algorithms. Here, an improvement of traditional methods is suggested which gives the better quality of recommendations. In the proposed method, the similarity between items is taken into consideration along with the contextual information that are derived using singularity of the ratings provided by each user. Later, comparison of the proposed algorithm is carried out with traditional algorithms, proving its efficiency.

1.1.6 Organization of Thesis

- Chapter 2, gives the literature survey that includes the review of works of existing algorithms to find similarity.
- Chapter 3, deals with the description of proposed algorithm.
- In chapter 4, results of various implemented algorithms and proposed algorithms are discussed.
- Chapter 5, presents the conclusion drawn from the results as well as the scope for future work is discussed.

Chapter 2

Literature Survey

An extensive research is been done in the field of recommender systems and number of methods have come up, each having its advantages and disadvantages.

Sarwar *et al.* [1] proved that the item-item scheme provides better quality of predictions than user-user scheme. The improvement in quality is consistent over different neighbourhood size. Another observation is that the item neighbourhood is fairly static, which can be potentially pre-computed, which results in very high online performance.

Bobadilla *et al.* [2] proposed that recommender systems contains information that are not used by traditional metrics, but singularity based approach provides a method to use those information thereby increasing the accuracy of similarity measurement techniques. The similarities are computed providing a weight to each rating, *i.e.*, singularity. More singular items should have high value in similarity computation as compared to items that are less singular.

Choi *et al.* [3] described that traditional systems use only similarity between users, irrespective of similarity between items. But, if the similarity between users for a target item is calculated, taking into consideration the similarity of the target item with other items, then the accuracy of the recommender system was seen to be improved.

2.1 Methods of Similarity calculation

Number of methods are available for similarity computation. Some of them are:

2.1.1 Pearson Correlation

In this similarity measure, similarity is found between any two items i and j keeping in mind that a particular user has rated both of these items. Advantage of this approach is that calculation is not done for all the users, conditions where customers have rated both the items i and j are only evaluated[1].

$$similarity(i, j) = \frac{\sum_{u \in \mathbf{U}} (R_{u,i} - \bar{R}_i)(R_{u,j} - \bar{R}_j)}{(\sqrt{\sum_{u \in \mathbf{U}} (R_{u,i} - \bar{R}_i)^2} \sqrt{\sum_{u \in \mathbf{U}} (R_{u,j} - \bar{R}_j)^2})} \quad (2.1)$$

Here, $R_{u,i}$ is rating provided by user u to item i . \bar{R}_i is Average of the ratings provided to item i .

2.1.2 Cosine-based Similarity

The concept of angle is used here to calculate the similarity among the different items [1]. The similarity between the two items is calculated by finding out the cosine of the angle between them. Formally, in the $n \times m$ rating matrix (that is user-item matrix), similarity between any pair of items is, denoted by

$$similarity(i, j) = \cos(\theta) = \frac{I \cdot J}{\|I\|^2 \|J\|^2} \quad (2.2)$$

I and J are the column vectors of ratings of item i and item j respectively.

Its is simple to evaluate. It gives the value in between $[0,1]$. The variation in the ratings given to the items between the different users is not taken for the computation.

2.1.3 Adjusted Cosine

In this similarity measure, the difference in the rating scale between different users is taken into account by subtracting the average rating of user from each co-rated pair [1].

$$similarity(i, j) = \frac{\sum_{u \in \mathbf{U}} (R_{u,i} - \bar{R}_u)(R_{u,j} - \bar{R}_u)}{(\sqrt{\sum_{u \in \mathbf{U}} (R_{u,i} - \bar{R}_u)^2} \sqrt{\sum_{u \in \mathbf{U}} (R_{u,j} - \bar{R}_u)^2})} \quad (2.3)$$

2.1.4 Similarity based on singularities

Here, similarity between users $u1$ and $u2$ is calculated taking into consideration the singularity of items [2]. This technique, provides the benefit that could lower the weightage of item that has generally been rated high or low.

$$similarity(u1, u2) = \frac{k1 + k2 + k3}{3} \quad (2.4)$$

where,
 $k1, k2, k3$ are defined as,

$$\begin{aligned} k1 &= \frac{\sum_{i \in \mathbf{A}} (1 - (R_{u1,i} - R_{u2,i})^2)(s_p^i)^2}{|A|} \\ k2 &= \frac{\sum_{i \in \mathbf{B}} (1 - (R_{u1,i} - R_{u2,i})^2)(s_p^i)(s_n^i)}{|B|} \\ k3 &= \frac{\sum_{i \in \mathbf{C}} (1 - (R_{u1,i} - R_{u2,i})^2)(s_n^i)^2}{|C|} \end{aligned}$$

2.1.5 Similarity between users taking item based similarity as weight

Here, similarity between users $u1$ and $u2$ is calculated, but using the similarity between the items as a weight to it [3]. This approach

gives better results, as the similarity between users is calculated keeping in mind the similarity between items as well.

$$similarity(u1, u2) = \frac{\sum_{j=1}^m t1^2 * t2 * t3}{\sqrt{\sum_{j=1}^m (t1 * t2)^2} \sqrt{\sum_{j=1}^m (t1 * t3)^2}} \quad (2.5)$$

where,
 $t1, t2, t3$ are defined as,

$$t1 = Isim(i, j)$$

$$t2 = (R_{u1,j} - \bar{R}_{u1})$$

$$t3 = (R_{u2,j} - \bar{R}_{u2})$$

Here, $Isim(i, j)$ is the similarity between items i and j .

2.2 Methods of prediction calculation

After calculating similarity between items, the prediction can be calculated using following methods [1, 2].

$$P_{u,i} = \frac{\sum_{j=1}^k similarity(i, j) * R_{u,j}}{|\sum_{j=1}^k similarity(i, j)|} \quad (2.6)$$

Here prediction of item i for any user u is calculated by taking into consideration other items that the particular user has rated, and how similar are those items to the item for which prediction is to be done.

$$P_{u,i} = \bar{R}_u + \frac{\sum_{j=1}^k similarity(i, j) * (R_{u,j} - \bar{R}_j)}{|\sum_{j=1}^k similarity(i, j)|} \quad (2.7)$$

Here, prediction of item i for any user u is calculated by taking into consideration the other items, that the particular user has rated,

and how similar are those items to the item for which prediction is to be done. Also, this method considers the average rating that the item has got by all the users, and on an average what is the rating that the user under consideration gives to the items.

Chapter 3

Proposed Work

Singularity-based approach uses the contextual information present, which other collaborative filtering algorithms ignore. In this approach singularity of the ratings given by each user is calculated. If a user gives high rating to all the items or low ratings to all the items (*i.e.*, same rating to all the items) then considering that user for calculation of similarity between items will not be beneficial; whereas, if a user has rated only two items differently to the rest of items, then similarity between those two items can be calculated easily. The singularity information obtained can be combined as a weight while calculating the similarity between items, thereby lessening the worth of user who rates almost all items similarly. This method is based on the hypothesis that value of similarity must be modulated by the value of singularity, in such a way that very singular similarity should be given a higher value.

The ratings provided by users are categorised into relevant rating *i.e.*, rating ≥ 4 and non-relevant rating *i.e.*, rating < 4 . Now, in calculation of similarity between two items a user can rate both the item as relevant (case A), one item as relevant and other as non-relevant (case B), both items as non-relevant (case C). Taking all the 3 cases into consideration we have to calculate the similarity between items, and accordingly apply the value of singularity to it. Here, U is set of all the users and N is set of all the items. T is the total number of items rated by a particular user. P_x and N_x are the number of positive and negative ratings provided by the user x .

Algorithm 1 Modified Singularity-based Collaborative Filtering

Input : Rating matrix**Output :** Prediction matrix

```
1: procedure MODIFIED_SIGULARITY
2:   for all users  $x \in U$  do
3:
4:      $S_p^x \leftarrow 1 - \frac{P_x}{T}$  (3.1)
5:      $S_n^x \leftarrow 1 - \frac{N_x}{T}$  (3.2)
6:   end for
7:   for all items  $i \in N$  do
8:     for all items  $j \in N$  do
9:        $d1 \leftarrow \frac{\sum_{x \in \mathbf{A}} (1 - (R_{x,i} - R_{x,j})^2) (s_p^x)^2}{|A|}$ 
10:       $d2 \leftarrow \frac{\sum_{x \in \mathbf{B}} (1 - (R_{x,i} - R_{x,j})^2) (s_p^x) (s_n^x)}{|B|}$ 
11:       $d3 \leftarrow \frac{\sum_{x \in \mathbf{C}} (1 - (R_{x,i} - R_{x,j})^2) (s_n^x)^2}{|C|}$ 
12:       $similarity(i, j) \leftarrow \frac{d1 + d2 + d3}{3}$  (3.3)
13:     end for
14:   end for
15:   for all users  $u \in U$  do
16:     for all items  $i \in N$  do
17:       if  $R_{u,i} = 0$  then
18:          $P_{u,i} \leftarrow \frac{\sum_{j=1}^k similarity(i, j) * R_{u,j}}{|\sum_{j=1}^k similarity(i, j)|}$ 
19:       or
20:          $P_{u,i} \leftarrow \bar{R}_u + \frac{\sum_{j=1}^k similarity(i, j) * (R_{u,j} - \bar{R}_j)}{|\sum_{j=1}^k similarity(i, j)|}$ 
21:       end if
22:     end for
23:   end for
24: end procedure
```

Chapter 4

Results and Analysis

4.1 Evaluation Metrics

- **Mean Absoulte Error:**MAE is a measure of deviation of recommendations from their true user-specific value. For each rating-prediction pair $\langle P_i, Q_i \rangle$ this metric treats the absolute error between them , i.e, $|P_i - Q_i|$ equally. The MAE is computed by first summing these absolute errors of the N corresponding ratings-prediction pairs and then computing the average [1, 2, 3, 7]. Formally,

$$\text{MAE} = \frac{\sum_{i=1}^N |P_i - Q_i|}{N} \quad (4.1)$$

The lower the MAE,the more accurately the recommendation engine predicts user ratings.

- **Mean Absoulte Error for good items:**Mean absolute error is calculated only for those items which have positive ratings in the test set.
- **Root Mean Square Error:**It is a metric represented by the square root of the average of the squares of the differences between actual and estimated preference values. [7]

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^N (P_i - Q_i)^2}{N}} \quad (4.2)$$

- **Precision and recall:** The precision is the proportion of recommendations that are good recommendations, and recall is the proportion of good recommendations that appear in top recommendations [2, 3, 7].
- **F1-score:** The F-Score or F-measure is a measure of a statistic test's accuracy. It considers both precision and recall measures of the test to compute the score. We could interpret it as a weighted average of the precision and recall, where the best F1 score has its value at 1 and worst score at the value 0 [2, 3, 7].

$$F1 = \frac{2 * precision * recall}{precision + recall} \quad (4.3)$$

4.2 Data-sets Used

Movie lens datasets are used. It contains ratings in the scale of 1-5.

- 100K data : The full data set consists of 100000 ratings by 943 users on 1682 items. Each user has rated at least 20 movies. Users and Items are numbered consecutively from 1. 80% of the ratings are used as the training set and rest 20% as the test set.
- 1M data: The full dataset contains 1,000,209 anonymous ratings of 3,952 movies made by 6,040 Movie Lens users who joined Movie Lens in 2000. 80% of the ratings are used as the training set and rest 20% as the test set.

4.3 Observations

4.3.1 Method 1 (without averaging)

Prediction calculation is done using:

$$P_{u,i} = \frac{\sum_{j=1}^k similarity(i,j) * R_{u,j}}{|\sum_{j=1}^k similarity(i,j)|}$$

Mean Absolute Error

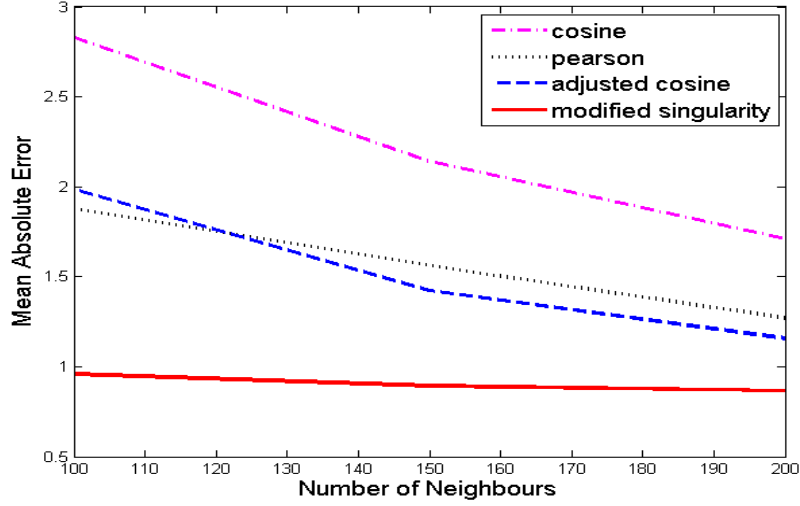


Figure 4.1: Mean absolute error vs number of neighbours for 100K data set.

Mean Absolute Error

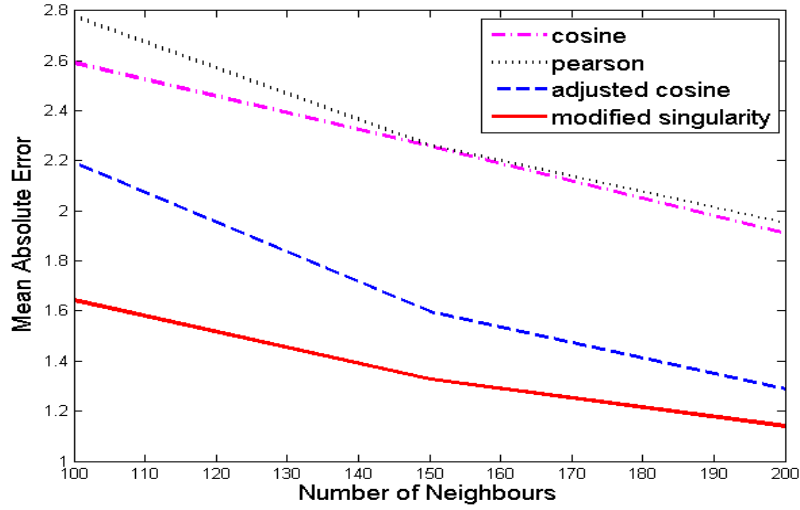


Figure 4.2: Mean absolute error vs number of neighbours for 1M data set.

- Mean absolute error was found to reduce as the number of neighbour increases. Among all the methods, modified singularity approach was found to have least MAE value with adjusted cosine, Pearson correlation and cosine approaches fol-

lowing it.

MAE For Good Items

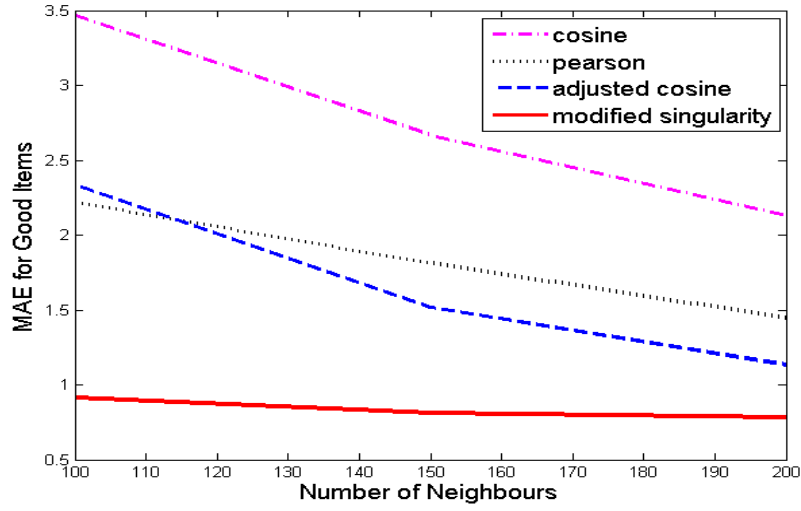


Figure 4.3: MAE for good items vs number of nieghbours for 100K dataset

MAE For Good Items

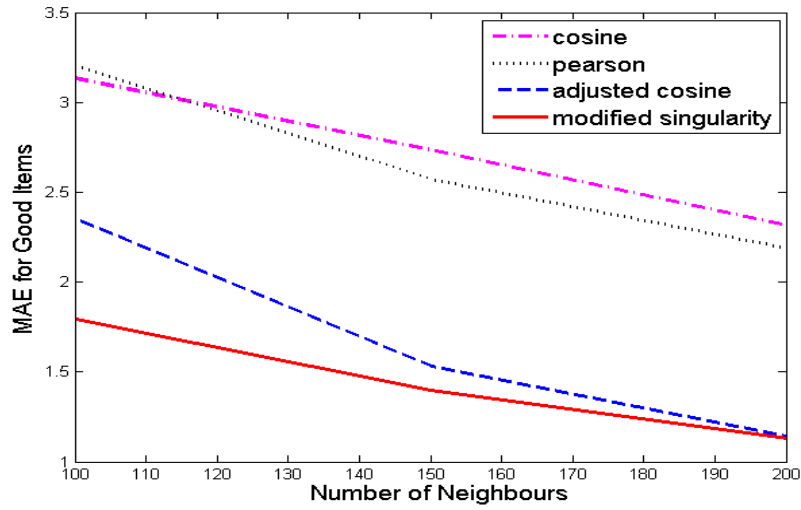


Figure 4.4: MAE for good items vs number of nieghbours for 1M dataset

- Here, only good items are considered to calculate the MAE values. The resulting graph shows less MAE for modified singularity method as compared to other three methods. Modified

Singularity method works well in the prediction for good items as well.

ROOT MEAN SQAURE ERROR

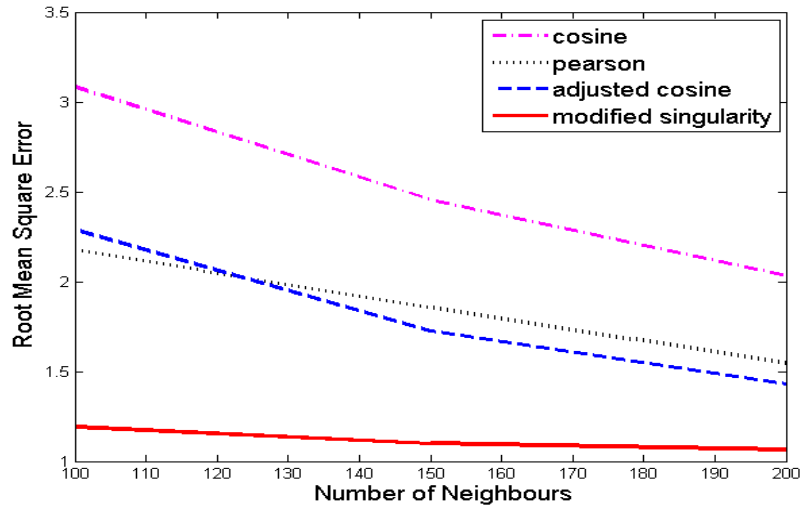


Figure 4.5: RMSE vs number of nieghbours for 100K dataset

ROOT MEAN SQAURE ERROR

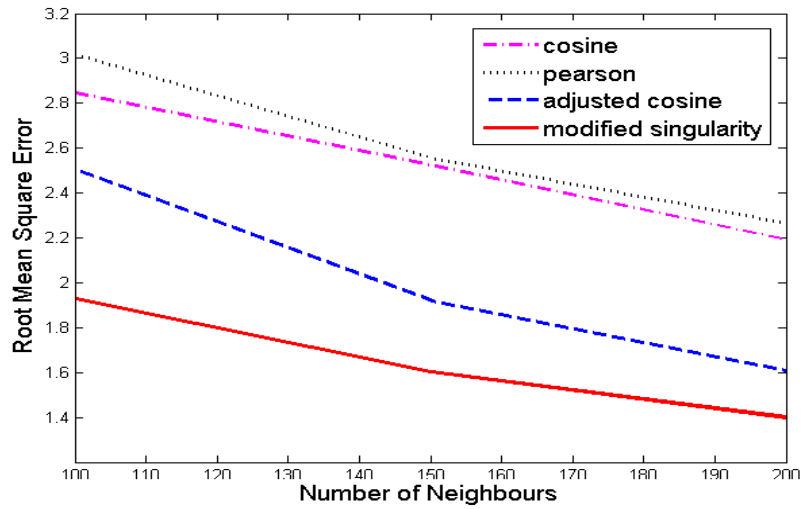


Figure 4.6: RMSE vs number of nieghbours for 1M dataset

- The root mean square value of singularity based approach was observed to be less than other methods , thereby proving the

efficiency of the process.

- Precision, recall and F1 score were computed and the observations are as follows:

PRECISION

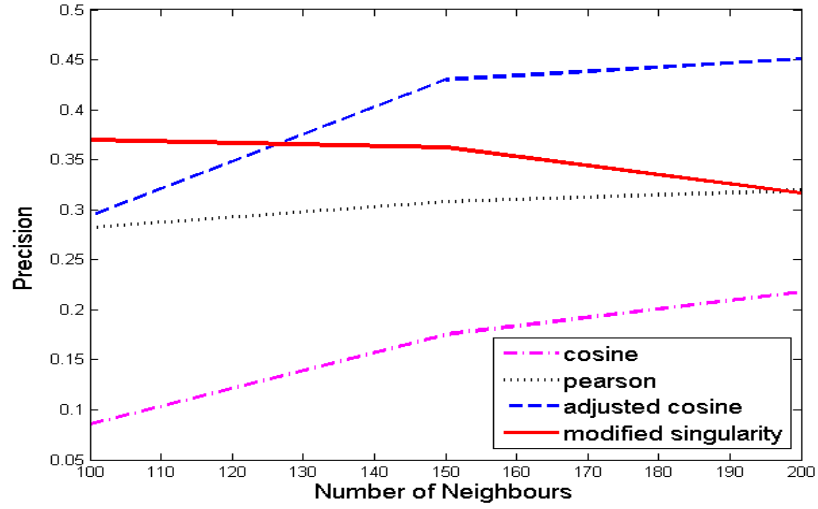


Figure 4.7: Precision vs number of nieghbours for 100K dataset

PRECISION

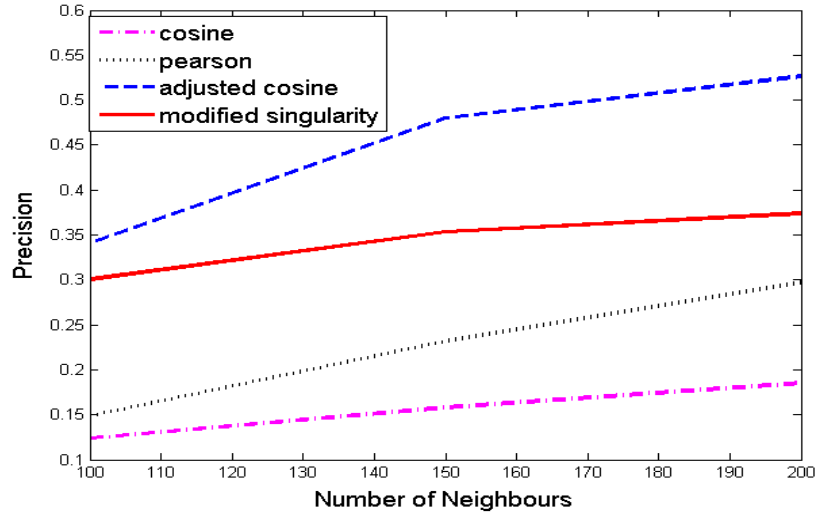


Figure 4.8: Precision vs number of nieghbours for 1M dataset

The precision of the adjusted cosine method was found to be better followed by modified singularity, Pearson and cosine approaches.

RECALL

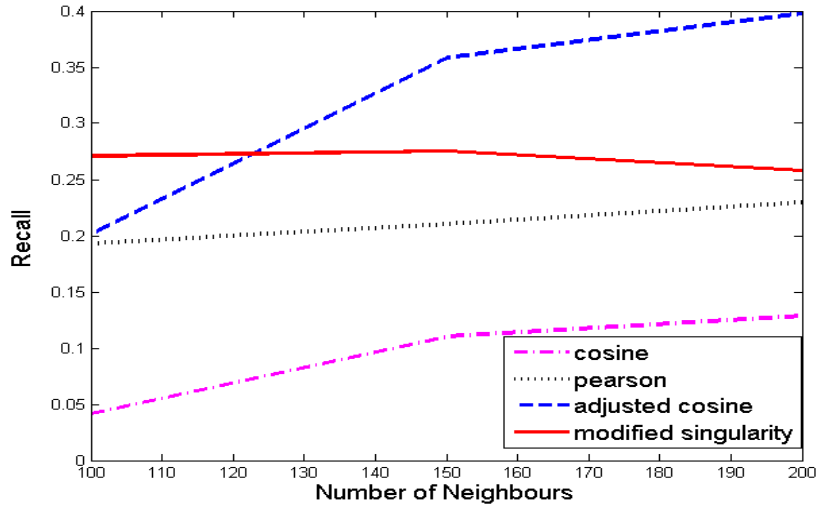


Figure 4.9: RECALL vs number of nieghbours for 100K dataset

RECALL

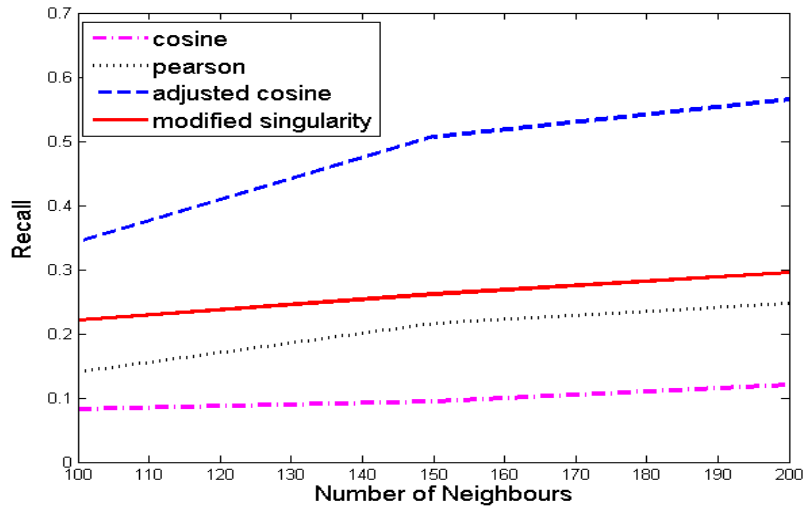


Figure 4.10: RECALL vs number of nieghbours for 1M dataset

Higher recall value of adjusted cosine indicated that more pro-

portion of good recommendations appear in top recommendations using this method followed by modified singularity, Pearson and cosine approaches.

F1-SCORE

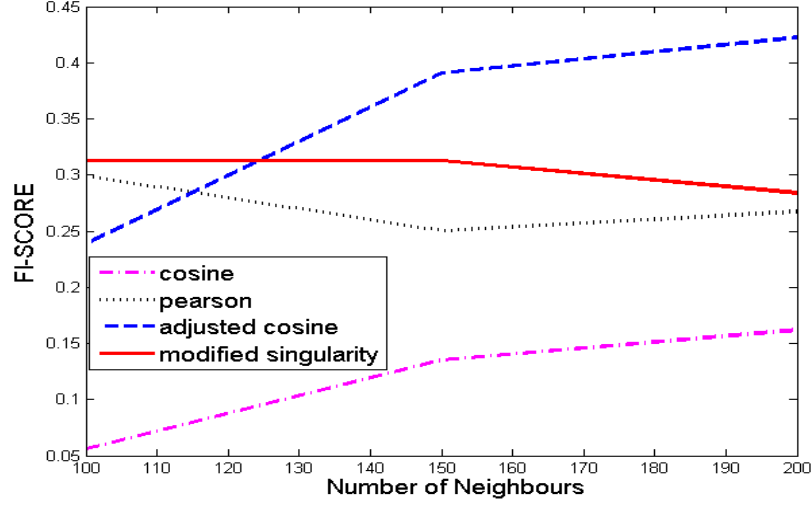


Figure 4.11: F1-score vs number of nieghbours for 100K dataset

F1-SCORE

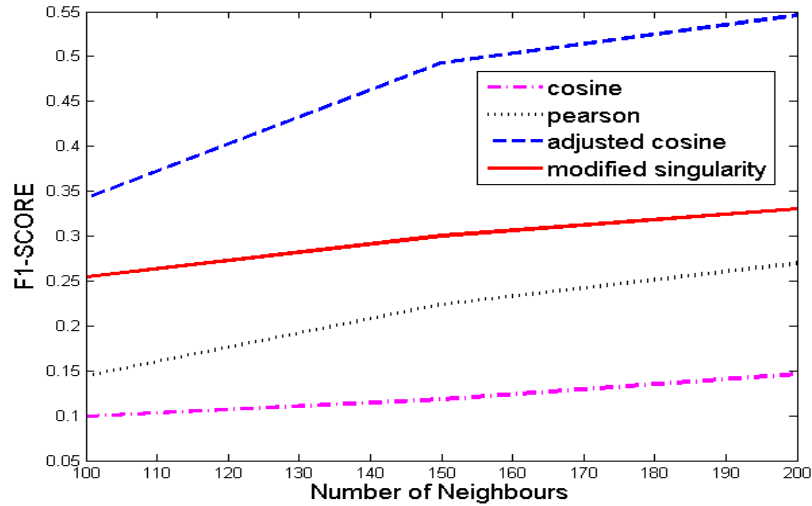


Figure 4.12: F1-score vs number of nieghbours for 1M dataset

F1-score of adjusted cosine technique was found to be far bet-

ter than the modified singularity method and Pearson and cosine following it.

4.3.2 Method 2 (with averaging)

We calculate prediction by:

$$P_{u,i} = \bar{R}_u + \frac{\sum_{j=1}^k \text{similarity}(i,j) * (R_{u,j} - \bar{R}_j)}{|\sum_{j=1}^k \text{similarity}(i,j)|}$$

Mean Absolute Error

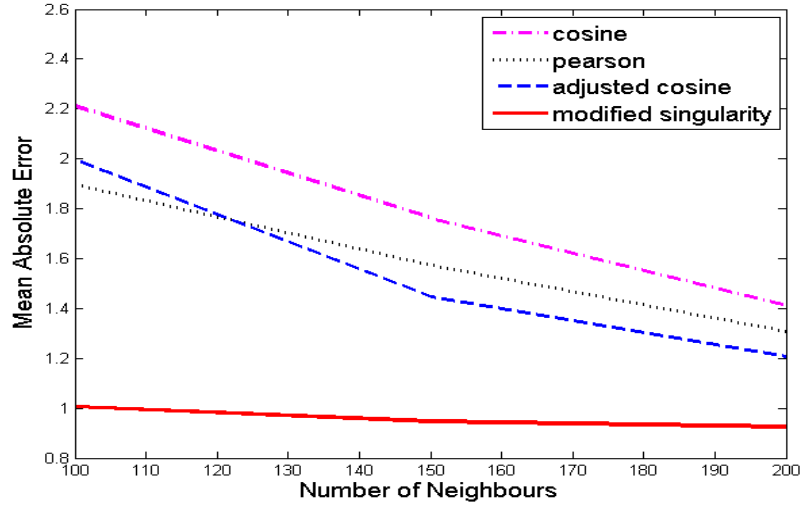


Figure 4.13: Mean absolute error vs number of neighbours for 100K data set.

Mean Absolute Error

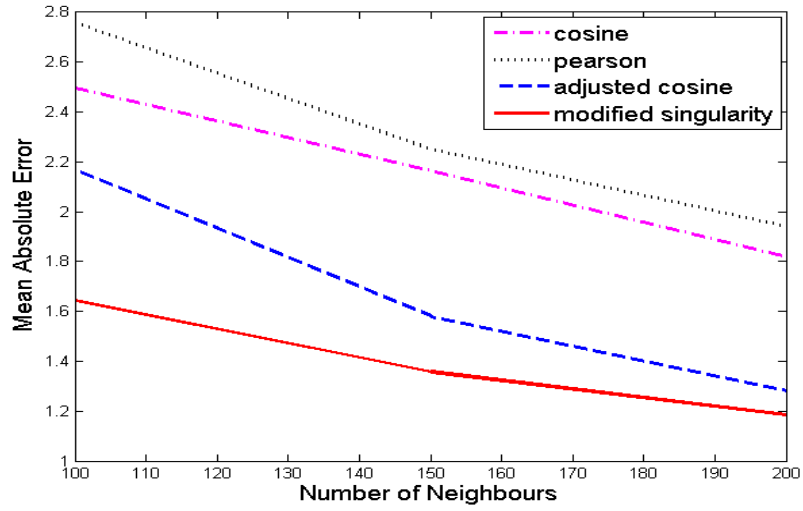


Figure 4.14: Mean absolute error vs number of neighbours for 1M data set.

- Mean absolute error was found to reduce as the number of neighbour increases. Among all the methods, modified singularity approach was found to have least MAE value with adjusted cosine, Pearson correlation and cosine approaches following it.
- For good items singularity based approach gave better results as compared to other approaches.

MAE For Good Items

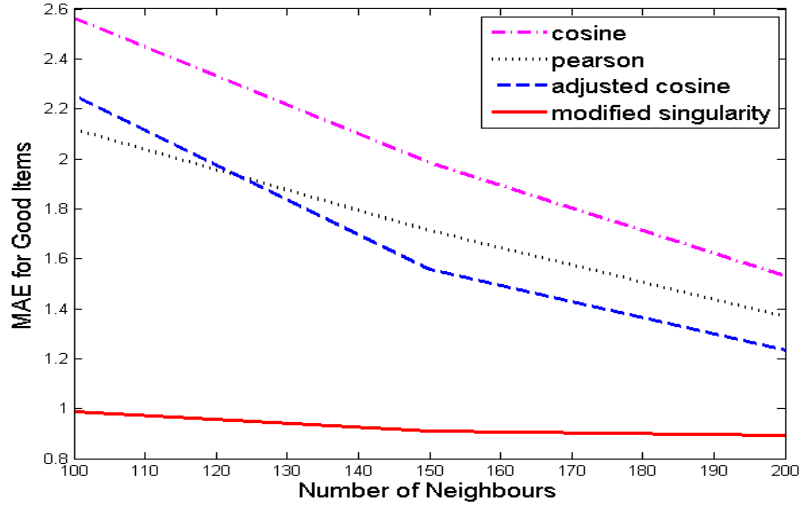


Figure 4.15: MAE for good items vs number of nieghbours for 100K dataset

MAE For Good Items

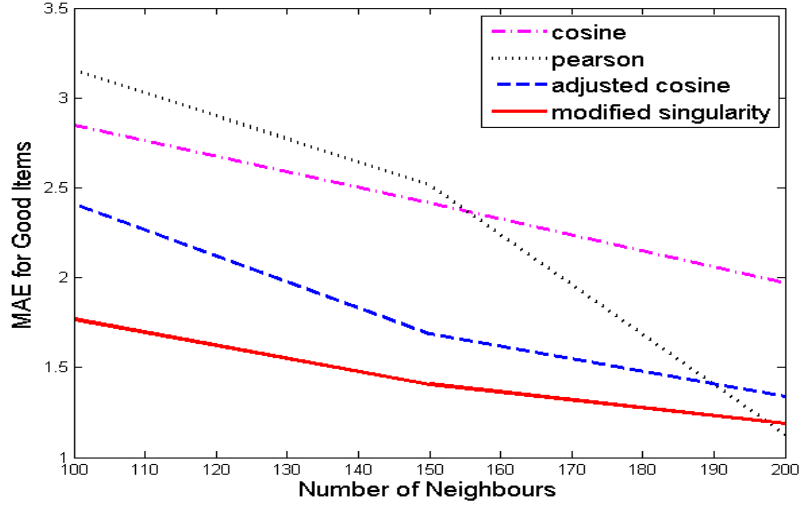


Figure 4.16: MAE for good items vs number of nieghbours for 1M dataset

Here, only good items are considered to calculate the MAE values. The resulting graph shows less MAE for modified singularity method as compared to other three methods. Modified Singularity method works well in the prediction for good items

as well.

ROOT MEAN SQAURE ERROR

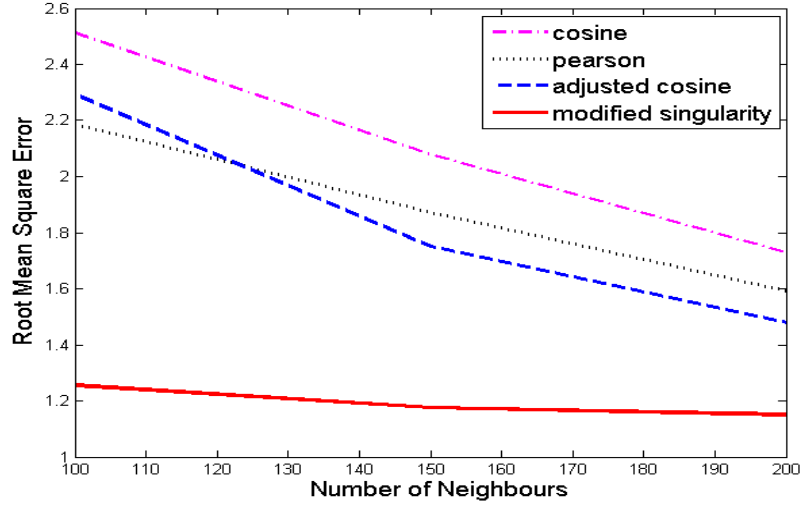


Figure 4.17: RMSE vs number of nieghbours for 100K dataset

ROOT MEAN SQAURE ERROR

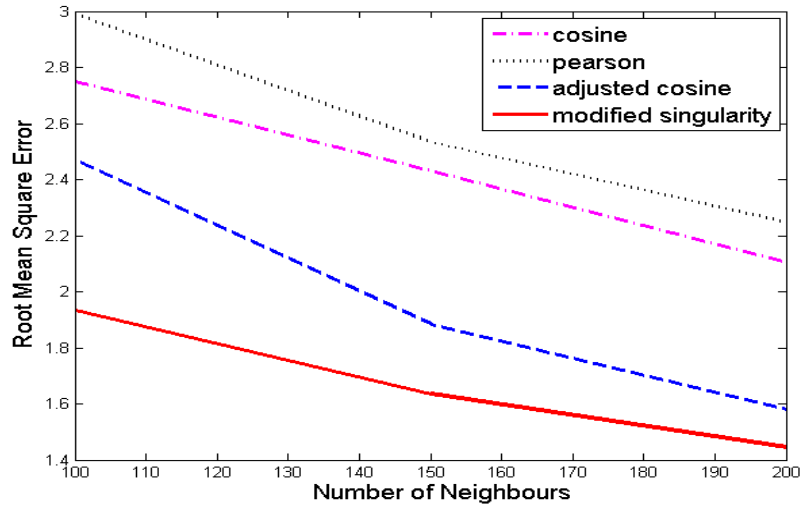


Figure 4.18: RMSE vs number of nieghbours for 1M dataset

- The root mean square value of singularity based approach was observed to be less than other methods , thereby proving the efficiency of the process.

- Precision, recall and F1 score were computed and the observations are as follows:

PRECISION

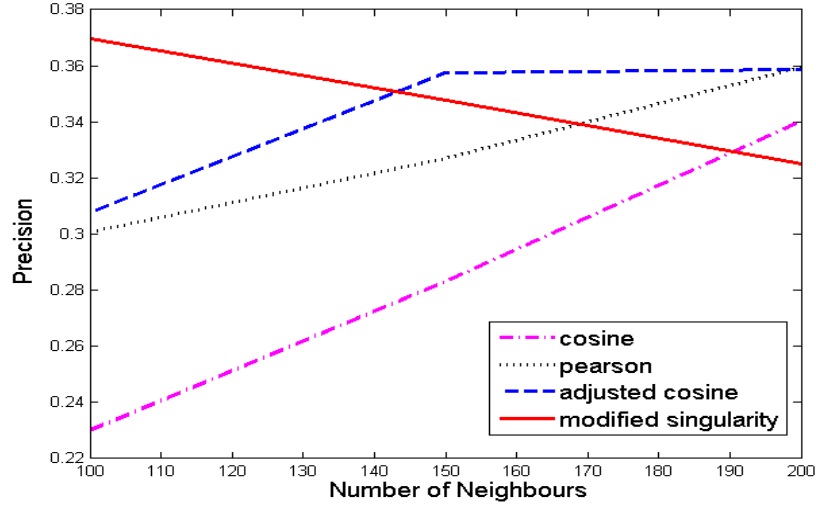


Figure 4.19: Precision vs number of nieghbours for 100K dataset

PRECISION

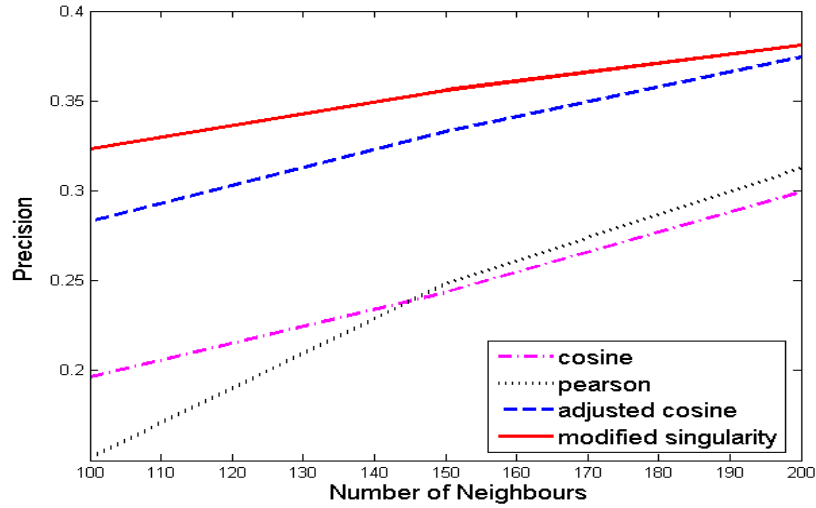


Figure 4.20: Precision vs number of nieghbours for 1M dataset

Modified singularity method showed the higher value of precision when 1M dataset is used thereby proving that more pro-

portion good of recommendations appear in top recommendations. Other three methods showed recall lower than the proposed method.

RECALL

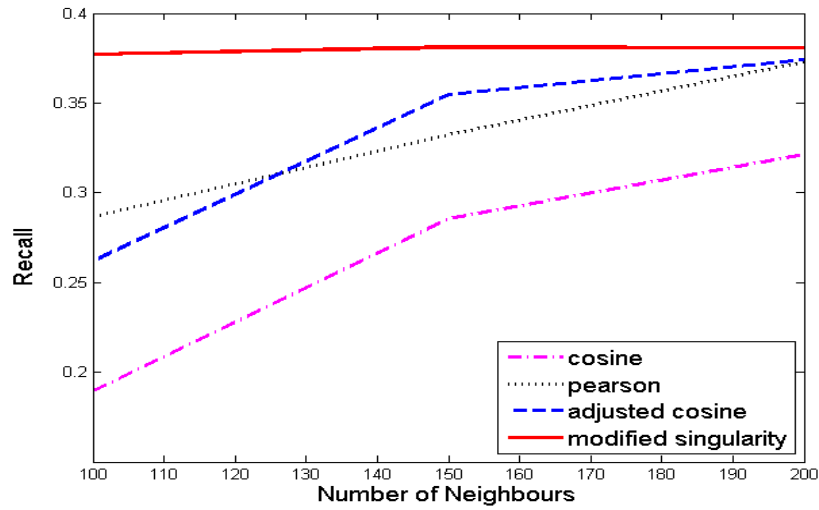


Figure 4.21: RECALL vs number of nieghbours for 100K dataset

RECALL

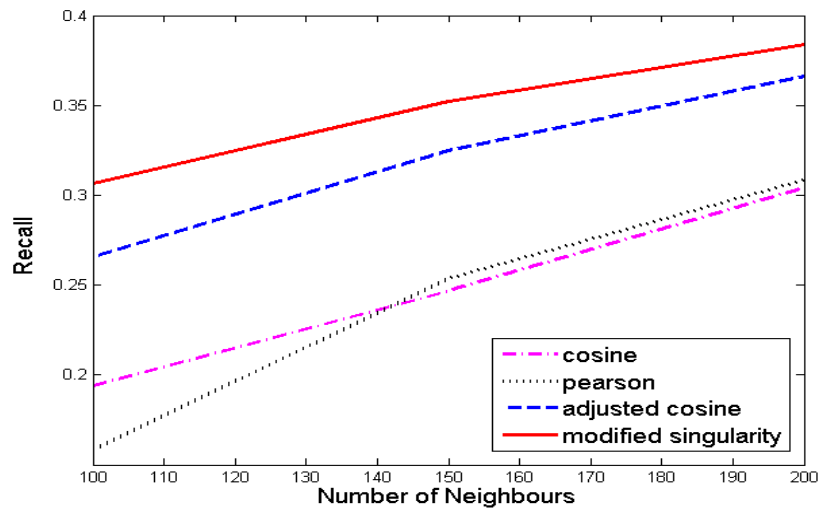


Figure 4.22: RECALL vs number of nieghbours for 1M dataset

Modified singularity method showed the higher value of recall

thereby proving that more proportion of recommendations is good recommendations. Other three methods showed precision lower than the proposed method.

F1-SCORE

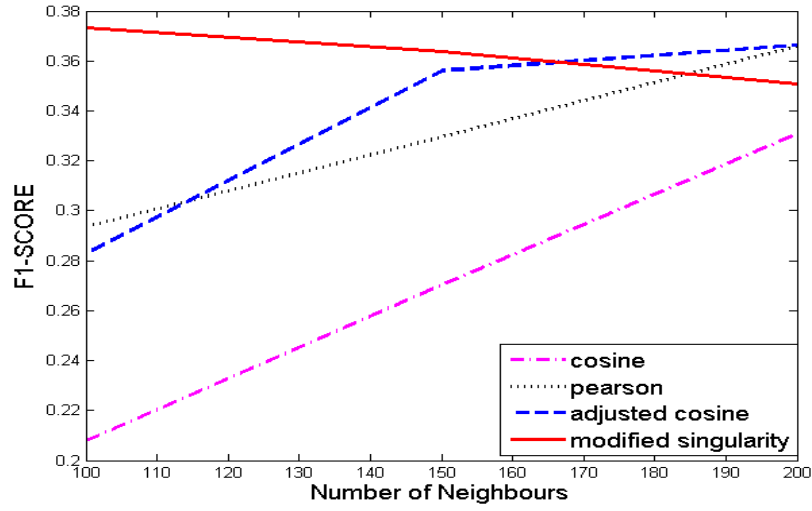


Figure 4.23: F1-score vs number of nieghbours for 100K dataset

F1-SCORE

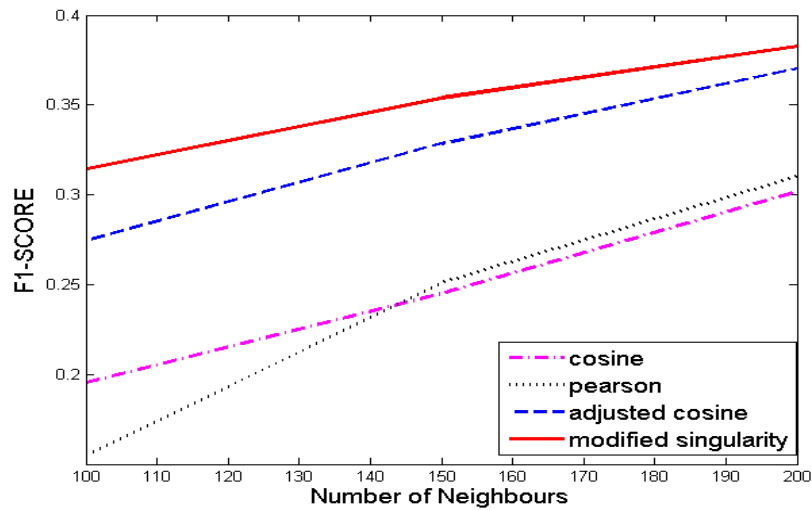


Figure 4.24: F1-score vs number of nieghbours for 1M dataset

Modified singularity method showed higher F1 score when 1M

dataset was used, thereby proving the efficiency of the method as compared to other three methods that showed lesser F1-score as compared to it.

4.4 Discussion

The results obtained from the proposed similarity measure (singularity-based) are improved vastly as compared to tradition similarity algorithms, *i.e.*, Pearson correlation, cosine-based, adjusted cosine. Improvements are especially noticeable in the mean absolute error, root mean square error, MAE for good items. In both the methods of prediction calculation used, with both the datasets, these parameters were far better as compared to traditional similarity metrics. F1-score of the proposed algorithm was observed to be similar to the adjusted cosine or even worse in certain cases, adding a drawback to the proposed similarity calculation technique. An improvement in F1-score was observed when 1M data set was used and prediction calculation with averaging was done. Precision for the proposed technique was found to be better as compared to other techniques when prediction calculation with averaging was used , but using prediction calculation without averaging precision of adjusted cosine was observed to be better. Recall of the adjusted cosine based approach was found to be better in almost all the cases.

Chapter 5

Conclusion and Future Scope

From the above observations, it can be concluded that, MAE, MAE for good items and RMSE for singularity based approach was found to be better than other approaches in all the cases. F1-score for singularity based approach was found comparative to or in some cases poor than adjusted cosine based similarity measure, except when using prediction calculation with averaging, where it was observed to perform better. As a future work, one can try to improve the F1-score, and also this algorithm can be implemented using Hadoop cluster (using map-reduce programming) as it requires high computation to be done. Also, appropriate division of ratings into positive and negative set can be studied to get better results.

Bibliography

- [1] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. “*ItemBased Collaborative Filtering Recommendation Algorithms*”. Proceedings of the 10th international conference on World Wide Web. Pages 285-295, 2001.
- [2] Jesus Bobadilla, Fernando Ortega, Antonio Hernando. “*A collaborative filtering similarity measure based on singularities*”. Information Processing and Management, Volume 48, Issue 2. Pages 204-217, 2012.
- [3] Keunho Choi, Yongmoo Suh. “*A new similarity function for selecting neighbors for each target item in collaborative filtering*”. Knowledge-Based Systems. Pages 146-153, 2013.
- [4] M. Deshpande and G. Karypis. “*Item-based top-n recommendation algorithms*”. ACM Trans. Inf. Syst., 22(1). Pages 143-177, 2004.
- [5] Maddali Surendra Prasad Babu, and Boddu Raja Sarath Kumar. “*An Implementation of the User-based Collaborative Filtering Algorithm*” . (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 2 (3). Pages 1283-1286, 2011.
- [6] Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., and Riedl, J. “*GroupLens: An Open Architecture for Collaborative Filtering of Netnews*”. In Proceedings of CSCW, Chapel Hill, NC. 1994
- [7] Herlocker, J., Konstan, J.A., Terveen, L., Riedl, J. “*Evaluating collaborative filtering recommender systems*”. ACM Transactions on Information Systems (TOIS)-2004, 22.
- [8] Rong J., Joyce Y. and Luo S. “*An automatic weighting scheme for collaborative filtering*”. Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval. Pages 1-8, 2004.

- [9] D. Goldberg, D. Nichols, B. M. Oki, and D. Terry. “*Using collaborative filtering to weave an information tapestry*”. Communications of ACM, vol. 35, no. 12, pp. Pages 61-70, 1992.
- [10] Manos Papagelis and Dimitris Plexousakis . “*Qualitative analysis of user based and item-based prediction algorithms for recommendation agents*”. Engineering Applications of Artificial Intelligence. Pages 781-789, 2005.
- [11] Yehuda Koren. “*Factorization meets the neighbourhood: a multi faced collaborative filtering model*”. Proceedings of the 14th ACM SIGKD. Pages 426-434, 2008.